IDENTIFICATION

PRODUCT CODE:	MAINDEC-08-D1EC-D
PRODUCT NAME:	PDP-8, 8/I EXTENDED MEMORY CHECKERBOARD
DATE CREATED:	NOVEMBER 1, 1971
MAINTAINER:	DIAGNOSTIC GROUP
AUTHOR:	J. RICHARDSON - J. VROBEL

1. ABSTRACT

The PDP-8, 8/I Extended Memory Checkerboard diagnostic is designed to provide worst case half-select noise conditions in order to determine the operational status of core memory. Four data patterns, and their complements, are written and checked for error. The patterns provided will generate the worst case noise conditions for a PDP-8 or 8/I equipped with standard or specially purchased core stacks, and will test systems equipped with from 8K to 32K words of core memory. Automatic program relocation is provided in order to test all memory stacks from each stack.

Teletype print-outs are provided for error identification. Also, the operator is given a degree of control over the program by various SR settings. These are explained in detail in Section 8.2.

2. REQUIREMENTS

2.1 Equipment

A standard PDP-8 or 8/I equipped with at least 8K words of core memory.

2.2 Storage

The program occupies locations 0010 to 3334.

2.3 Preliminary Programs

The Binary Loader must be in memory. Also, all diagnostics for a basic 4K PDP-8 must have been previously run successfully.

3. LOADING PROCEDURE

a. Turn off the Teletype reader.

b. Set the SR to 7777.

c. Press LOAD ADDRESS; then START.

d. Place the Binary tape in Teletype reader and turn on the reader.

e. When the program has been loaded, stop the computer, turn off the reader, and remove the tape.

4. STARTING PROCEDURE

4.1 Starting Address

Start from address 200 to specify the amount of core memory to test; SR settings, and to receive a header sprint-out.

4.2 Restarting Address

Start from address 207 to change the test limits; SR settings, and to inhibit the header print-out.

4.3 Operator Action

Immediately after starting from address 200 or 207, the program will print TEST LIMITS. The operator must then specify, via the Teletype keyboard, the amount of core memory to test, followed by a carriage return.

The following rules govern the amount of memory to test:

a. Type two octal numbers, separating the numbers with a comma. The first number signifies the lowest order 4K stack to test; the second signifies the highest order.

b. The program expects the 4K stacks to be numbered sequentially starting with a stack 0.

c. If the highest order stack to test is typed as the first stack, the program will interchange the two values so as to make the second value the first to test.

d. After typing the second octal number, press the carriage return key to terminate the line.

e. The program will test the lowest and highest order 4K stack specified, plus every stack between, starting with the lowest specified.

f. Any single stack, or two or more sequential stacks may be specified.

g. The stack containing the program may be included when specifying two or more stacks. The stack containing the program will be tested after automatic program relocation takes place (see Section 5.3.1).

h. If a typing error is made, press the RUB-OUT key. TEST LIMITS will be printed again. All previous input is disregarded.

For the following examples assume the program to be located in stack 0, and the program has been started from address 200 or 207. The amount of core memory available is 32K.

Example A:

TEST LIMITS

0,7↵ (↵ denotes carriage return)

Example A indicates stacks 0, 1, 2, 3, 4, 5, 6 and 7 will be tested.

Example B:

TEST LIMITS
7,0↵

The program will perform exactly as Example A.

Example C:

TEST LIMITS
4,5↵

Only stacks 4 and 5 will be tested.

Example D:

TEST LIMITS
3,3↵

Stack 3 alone will be tested.

Example E:

TEST LIMITS
0,0 PROGRAM IS LOCATED IN FIELD 0
TEST LIMITS
0,1↵

Example E shows the message printed by the program when a single stack is selected which currently contains the program. TEST LIMITS is printed again, and the operator must then correct the test limits.

Operation of the program is unpredicatable if the amount of memory selected for testing exceeds the actual amount available, i.e., selecting 32K for testing on a PDP-8 or 8/I equipped with a maximum of 28K.

4.3.1 Setup SR

After the test limit is specified, the program will print SETUP SR. For normal program operation, the SR must be set to equal 0000_8. Press the carriage return key after setting the SR to 0000. The program will then run until stopped by the operator. Normal program operation is defined as performing all four checkerboard patterns on all of available memory from every memory stack.

5. OPERATING PROCEDURE

5.1 Program and Operator Action

a. Load the program into stack 0 using the procedure described in Section 3.

b. Set the SR to 200; press LOAD ADDRESS, and then start.

c. The message TEST LIMITS will be printed. Specify the limits, via keyboard, as described in Section 4.3.

d. The message SETUP SR will be printed. Set the SR to 0000_8, and press the carriage return key.

e. The program will perform all four tests on all of core memory specified, after which, automatic program relocation takes place.

5.2 Operational Switch Settings

Normal operation of the program requires the SR set to 0000_8. Refer to Section 8.2, applications, for switch settings provided for trouble-shooting.

5.3 Subroutine Abstracts

5.3.1 Program Relocation

Program relocation is governed entirely by the amount of core memory selected for testing. Under certain conditions the program will not relocate at all, but will remain in the current 4K stack to perform the tests (see below). The program first relocates to the highest order 4K stack under test. From there it relocates to the next lower stack (after performing all four tests). The program keeps relocating to the next lower stack until it reaches the lowest order stack under test. The testing and relocation cycle is then repeated.

The contents of the entire 4K stack are relocated. This enables the RIM Loader, and any other information to be carried with the program.

The program provides a degree of protection for itself by recording the first error encountered in any stack. When a faulty stack is next in sequence to contain the program, the program will skip the faulty stack and relocate to the first lower order stack which is error-free. If all lower order stacks are faulty, program relocation will not take place. The tests will be run again from the current stack. Relocation will resume when an error-free stack is found.

Also, the program will not relocate if any of the conditions described below exist.

a. Only one 4K stack is selected for testing.

b. SR 9 is on a 1 to inhibit relocation (see Section 8.2.6).

The INSTRUCTION FIELD indicators will indicate the current stack containing the program.

5.3.2 The Checkerboard Patterns

Four test patterns, and their complements, are used to test memory. All memory stacks, except the one with the program, are tested with one pattern before the next test is executed.

Any one, or any combination, of the four tests may be run by placing one, or any combination, of SR 3, 4, 5, or 6 on a 1 after the message SETUP SR is printed. The test specified by the most significant switch on a 1 will be executed first. SR 3, 4, 5 and 6 all on a 0 will enable all tests to be run. SR 3= test 1; 4= test 2; 5= test 3; 6= test 4.

The following steps are performed by each of the four tests:

a. Write the pattern once in all stacks selected for testing; starting with the lowest order stack.

b. Select the lowest order stack and perform a read, complement data, write sequence once on each location, until all 4K has been complemented.

c. Repeat step b 31 more times. The stack will end up with the pattern originally loaded. No error checking has been performed as yet.

d. Read 4-word segments and complement each segment 4 times; then read each of the 4 words and check for error.

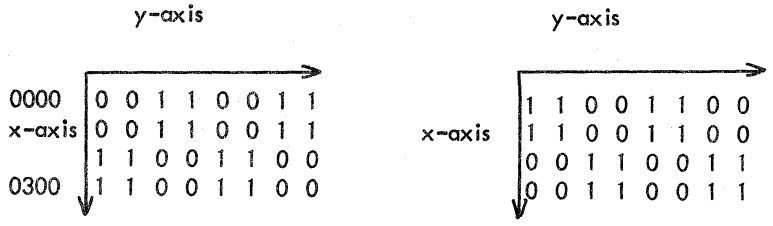
e. After checking the entire 4K stack for errors, repeat step d again. This time stall for a random period of time after reading and checking every 400_8 word block. The maximum stall is 18.4 ms; the minimum is 3 μs.

f. Setup for the next sequential 4K stack and repeat steps b through f.

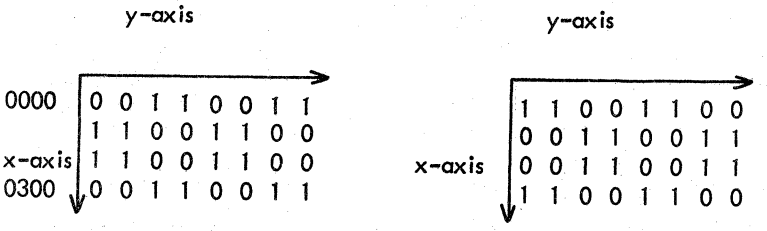
When all selected stacks have been checked the next test in sequence is executed, and steps a through f repeated. Program relocation takes place after the fourth test is executed in this manner.

The patterns generated by each test are shown below. The matrices represent portions of one bit plane.

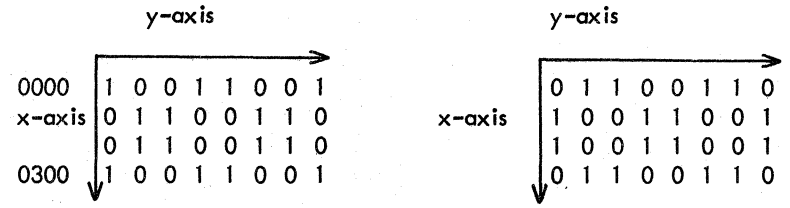
Test 1:



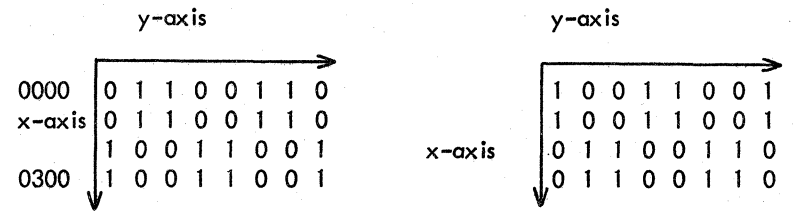
Test 2:



Test 3:



Test 4:



6. ERRORS

Starting the program from address 200 will give a header print-out after the SR has been setup. The header identifies the information printed when a data error is found. The header appears as:

FIELD OCTAL ADR. GOOD BAD TEST

Where:

FIELD	= an octal number (0 to 7) indicating the 4K field containing the error.
OCTAL ADR.	= the memory address which contains the incorrect data.
GOOD	= what the data in octal, should have been. This will always equal 0000 or 7777.
BAD	= the data as read. This will equal the good data except for one or more bits complemented.
TEST	= the number (1 to 4) of the test which detected the error.

After each error print-out the program continues on with the next sequential memory location.

6.1 Error Halts and Description

Placing SR 0 on a 1 during an error print-out will cause a halt at location 2641.
Press CONTINUE to resume testing.

7. RESTRICTIONS

7.1 Starting Restrictions

Start from address 200 to indicate the amount of core memory to test; to setup the SR and to receive a header print-out.

Starting from 207 requires the same operator action, but no header will be printed.

7.2 Operating Restrictions

None

8. MISCELLANEOUS

8.1 Execution Time

The time required to perform all four tests on one 4K memory stack is approximately 26 seconds.

8.2 Applications

For operating convenience, and as an aid to trouble-shooting, the SR may be used to control the program. The switch assignments and their effect on the program are described below. Please note that it is important that the program should be halted before changing the test selection switches. These switches are not sensed by the program during testing.

Halting the program with SR 0 is preferred, rather than with the STOP key. Using the STOP key may result in a halt while the program is in the process of relocating, which is disasterous.

8.2.1 Halt after Test or Error - SR 0

Placing SR 0 on a 1 at any time while the program is running will cause a halt after the current test is completed. The MB will equal 2461 in the current stack containing the program. Press CONTINUE to resume testing, or restart from 200 or 207 to enter new parameters.

Placing SR 0 on a 1 during an error type-out will also cause a halt at location 2461. Proceed exactly as described in the above paragraph.

8.2.2 Inhibit Error Print-out - SR 1

Placing SR 1 on a 1 causes all error print-outs to be inhibited. All other messages will not be inhibited. The program will continue to recognize errors, but will not print any information. SR 1 may be placed on a 1 or 0 while the program is running.

8.2.3 Bell on Error - SR 2

SR 2 on a 1 causes the program to ring the TTY BELL whenever an error is detected. This is convenient when testing with power supply margins. SR 2 has precedence over SR 1 if both should happen to be on a 1. SR 2 may be placed on a 1 or 0 while the program is running.

8.2.4 Test Selection SR 3 through 6

Any one, or any combination of tests may be executed by placing any one or any combination of SR 3 through 6 on a 1. Test selections may be made only when starting from 200 or 207. SR 3 specifies test 1; SR 4 test 2; SR 5 test 3; SR 6 test 4. The test specified by the most significant SR on a 1 will be executed first.

For most PDP-8s, SR 4 will provide the worst case pattern. For most PDP-8/Is, SR 5 will provide the worst case pattern.

If all four switches are on a 0, all four tests will be executed in order starting with test 1.

Program relocation is not effected, regardless of the SR settings.

8.2.5 Inhibit Program Relocation - SR 7

The program normally relocates automatically as indicated by the INSTRUCTION FIELD indicators. To retain the program in its current 4K field, place SR 7 on a 1 at any time. Changing SR 7 to a 0 will permit relocation to resume.

8.2.6 SR 8, 9 and 10 - Not Used

8.2.7 Change TEST LIMITS and SR - SR 11

Placing SR 11 on a 1 will cause the program to automatically restart from address 207. The TEST LIMITS and SR may then be changed. SR 11 is sensed only after all specified tests have been completed on all of memory under test.

8.2.8 Loop on Address

A subroutine is provided which may be used to continuously loop on a single location, or a group of consecutive locations. No error checking is performed. The routine performs a read, and immediately follows with a write, on each location. The loop time between two reads, or two writes, is approximately 22.5 μs.

Operating Procedure:

a. Set the INSTRUCTION FIELD switches to the current field, and the SR to 1700.

b. Set the DATA FIELD switches to equal the 4K field number to test.

c. Press LOAD ADDRESS.

d. Set the SR to equal the first address of the group.

e. Press START. A halt will occur at 1703. Set the SR to equal the last address of the group.

f. Press CONTINUE. The address(s) specified will be looped until stopped by the operator with STOP. SR 0 will not halt this routine.

To resume normal operation, restart the program from 200 or 207 of the current field.

9. PROGRAM DESCRIPTION

The PDP-8, 8/I Extended Memory Checkerboard diagnostic is designed to create worst case memory noise conditions on systems equipped with 8K to 32K words of memory. The program executes four checkerboard patterns, plus their complements, on each 4K memory field. In addition, the program automatically relocates from field to field in order to test all 4K fields from every 4K field. Under normal operation, the amount of core memory tested at one time is that specified by the operator minus the 4K field containing the program. A TTY keyboard input routine is provided to enable the operator to specify the exact number of 4K fields to be tested. A print-out is provided for each error detected by the program.

Further control of the program is given to the operator by means of the SR. The operator may halt the program, inhibit error print-outs, substitute the TTY BELL for error indication, halt after error print-out, select any one or a group of tests, inhibit program relocation, and create an automatic restart to change the amount of memory to test.

A small subroutine is provided which will continuously read and write any single, or a group of locations within any 4K field. The operator must specify the locations by means of the SR.

```
                        /
                        /PDP-8, 8I, 8S EXTENDED MEMORY CHECKBOARD TEST.
                        /COPYRIGHT 1971, DIGITAL EQUIPMENT CORP., MAYNARD, MASS.
                        /START AT 200, RESTART AT 211 TO SKIP HEADER.
                        /MIN. OF 8K OF CORE REQUIRED.
                        /
                0001    *1
                        /
5000 0001       5001            JMP .
   0 0002       0002            0002
     0003       0003            0003
                        /
                6201    CDF=6201
                6202    CIF=6202
                6214    RDF=6214
                6224    RIF=6224
     0004       0000            0
     0005       0000            0
     0006       0000            0
     0007       0000            0
4535 0010       0000            0
4535 0011       0000            0
2402 0012       0000            0
     0013       0000            0
     0014       0000    DATFLD, 0
     0015       0000    FLAGS,  0
     0016       0000    INSFLD, 0
     0017       0000    FLOAD,  0
     0020       0022    ERTBL,  ERWRD
     0021       0032    ENTBL,  ERWRD+10
     0022       0022    ERWRD,  .
     0023       7600            7600
     0024       7600            7600
     0025       7600            7600
     0026       7600            7600
     0027       7600            7600
     0030       7600            7600
     0031       7600            7600
     0032       7600            7600
0740 0033       0000    MCWA,   0
     0034       0010    K10,    10
 140 0035       0740    K740,   740
     0036       0400    K400,   400
     0037       0200    K200,   200
     0040       0100    K100,   100
     0041       0040    K40,    40
     0042       0020    K20,    20
     0043       2207    XLMTS,  SLMTS
     0044       0300    XTST1,  TST1
     0045       0400    XTST2,  TST2
     0046       0452    XTST3,  TST3
     0047       0600    XTST4,  TST4
     0050       3000    XMOVE,  CMOVE
     0051       2200    XSETU,  SETU1
     0052       7261    K261,   261
```

```
0053  0262  K262,   262
0054  0263  K263,   263
0055  0264  K264,   264
0056  7760  M20,    7760


0057  7740  M40,    7740
0060  7774  M4,     7774
0061  7773  M5,     7773
0262  0000  TNUM,   0
0063  1607  XBANK,  CBANK
0064  0652  X0011,  W0011
0065  0667  X1100,  W1100
0066  0704  X0110,  W0110
0067  0721  X1001,  W1001
0070  1600  XKBNK,  CKBNK
0071  1624  XTBNK,  NXTBNK
0072  0000  COUNT,  0
0073  0000  FLCNT,  0
0074  0000  LOOP,   0
0075  0736  XRALL,  RDALL
0076  1037  XCHK1,  RCHK1
0077  1054  XCHK1C, RCHK1C
0100  1071  XCHK2,  RCHK2
0101  1106  XCHK2C, RCHK2C
0102  1123  XCHK3,  RCHK3
0103  1140  XCHK3C, RCHK3C
0104  1200  XCHK4,  RCHK4
0105  1217  XCHK4C, RCHK4C
0106  1056  TDM20,  TAD M20
0107  1057  TDM40,  TAD M40
0110  4515  JMS1,   JMS I XRD1
0111  4516  JMS2,   JMS I XRD2
0112  4517  JMS3,   JMS I XRD3
0113  4520  JMS4,   JMS I XRD4
0114  4552  JMS5,   JMS I XSALL
0115  1245  XRD1,   RD1
0116  1322  XRD2,   RD2
0117  1400  XRD3,   RD3
0120  1455  XRD4,   RD4
0121  2000  XRROR,  ERROR
0122  0000  MEMADR, 0
0123  0000  FIRST1, 0
0124  0000  LAST1,  0
0125  6201  KCDF,   6201
0126  6202  KCIF,   6202
0127  2641  XHLT,   HALT
0130  0213  XRTN,   RTN1
0131  1646  XFILD,  FEILD
0132  2146  XPRER,  PRERR
0133  0007  K7,     7
0134  0000  CHAR,   0
0135  2474  XHDR,   PHDR
0136  2146  XPERR,  PRERR
0137  2115  XPING,  SPING
```

```
0140  7764    M14,    7764
0141  7770    M10,    7770
0142  0260    K260,   260
0143  0215    K215,   215
0144  0377    K377,   377
0145  0370    K370,   370
0146  0277    K277,   277
0147  2154    XCRLF,  CRLF
0150  0001    K1,     1
0151  0000    NXLOC,  0
0152  2702    XSALL,  STALL
0153  0000    EXIT,   0
0154  2166    LASTX,  LAST

      0200    *200
0200  6002    BEGIN,  IOF             /PI OFF
0201  7200            CLA
0202  3015            DCA FLAGS               /CLEAR PROGRAM FLAGS
0203  6224            RIF
0204  3016            DCA INSFLD
0205  4443            JMS I XLMTS     /SETUP TEST LIMITS
0206  4677            JMS I XSTSR     /SETUP SR
0207  4535            JMS I XHDR      /PRINT HEADER
0210  5213            JMP RTN1
                      /
                      /RESTART HERE
                      /
0211  4443    RSTRT1, JMS I XLMTS     /SET TEST LIMITS
0212  4677            JMS I XSTSR     /SETUP SR
0213  6224    RTN1,   RIF             /READ INSTRUCTION FIELD
0214  3016            DCA INSFLD      /CURRENT FIELD
0215  4531            JMS I XFILD
0216  1141            TAD M10                 /-10
0217  3074            DCA LOOP
0220  7600    ALAW,   7600
0221  1220            TAD ALAW
0222  2022            ISZ ERWRD
0223  3422            DCA I ERWRD
0224  2074            ISZ LOOP
0225  5220            JMP ALAW
0226  1220            TAD ALAW
0227  3554            DCA I LASTX
0230  1020            TAD ERTBL
0231  3022            DCA ERWRD
                      /
                      /EXAMINE SR
0232  1033            TAD MCWA
0233  0035            AND K740                /MADK 3,4,5 AND 6
0234  7440            SZA             /DO ALL IF 0
0235  5241            JMP EXAM1
0236  1033            TAD MCWA
0237  1035            TAD K740                /SET ALL TEST BITS
```

```
0240  3033          DCA MCWA                /SAVE
0241  7200  EXAM1,  CLA
0242  1033          TAD MCWA
0243  0036          AND K400
0244  7440          SZA                     /TEST 1 IF NO SKIP
0245  5444          JMP I XTST1
0246  7200  EXAM2,  CLA
0247  1033          TAD MCWA
0250  0037          AND K200
0251  7440          SZA                     /TEST 2 IF NO SKIP
0252  5445          JMP I XTST2
0253  7200  EXAM3,  CLA
0254  1033          TAD MCWA
0255  0040          AND K100
0256  7440          SZA                     /TEST 3 IF NO SKIP
0257  5446          JMP I XTST3
0260  7200  EXAM4,  CLA
0261  1033          TAD MCWA
0262  0041          AND K40
0263  7440          SZA                     /TEST 4 IF NO SKIP
0264  5447          JMP I XTST4
0265  4531          JMS I XFILD             /RESTORE DATA FIELD

0266  7604          LAS
0267  0150          AND K1
0270  7440          SZA                     /CHECK SR 11
0271  5211          JMP RSTRT1
0272  7604          LAS
0273  0042          AND K20
0274  7440          SZA                     /INHIBIT MOVE IF A 1
0275  5213          JMP RTN1
0276  5450          JMP I XMOVE             /GO RELOCATE
                    /
0277  2645  XSTSR,  SETSR


                    /
                    /TEST 1.  WRITE CHECKER PATTERN #1.
                    /
0300  4451  TST1,   JMS I XSETU             /SET DF TO 1ST FIELD
0301  7200          CLA
0302  1052          TAD K261
0303  3062          DCA TNUM                /TEST NUMBER
0304  7040          CMA
0305  3010          DCA 10                  /SET ADDRESS COUNT TO 7777
0306  4463          JMS I XBANK             /SEE IF FIELD HAS PROGRAM
0307  7410          SKP                     /NO,  BEGIN WRITING
0310  5322          JMP EXT1                /DONE ALL,  NOW READ ALL
0311  1345          TAD KXT1
0312  3153          DCA EXIT
0313  4464          JMS I X0011             /WRITE 0011
0314  4464          JMS I X0011             /WRITE 0011 64 TIMES
0315  4465          JMS I X1100             /WRITE 1100 128 TIMES
0316  4465          JMS I X1100
```

```
0317  5313          JMP .-4          /KEEP WRITING
0320  4471  XIT1,   JMS I XTBNK      /SETUP FOR NEXT FIELD
0321  5304          JMP TST1+4
                    /
0322  4476  EXT1,   JMS I XCHK1      /READ EACH FIELD AND CHECK
                                     /FOR ERRORS,
0323  5324          JMP TST1C                /NOW WRITE COMPLEMENT
                    /
                    /WRITE COMPLEMENT OF PATTERN 1
                    /
0324  4451  TST1C,  JMS I XSETU      /SEE DF TO 1ST FIELD.
0325  7240          CLA CMA
0326  3010          DCA 10           /SET ADDRESS COUNT TO 7777
0327  4463          JMS I XBANK      /SEE IF FIELD HAS PROGRAM
0330  7410          SKP
0331  5343          JMP EXT1C                /ALL DONE,  READ ALL
0332  1346          TAD KXT1C
0333  3153          DCA EXIT
0334  4465          JMS I X1100      /WRITE 1100
0335  4465          JMS I X1100      /WRITE 1100 16 TIMES
0336  4464          JMS I X0011
0337  4464          JMS I X0011      /WRITE 0011 128 TIMES
0340  5334          JMP .-4          /KEEP WRITING
0341  4471  XIT1C,  JMS I XTBNK      /SETUP FOR NEXT FIELD
0342  5325          JMP TST1C+1
                    /
0343  4477  EXT1C,  JMS I XCHK1C     /READ EACH BANK AND CHECK
                                     /FOR ERRORS,
0344  5246          JMP EXAM2                /SEE IF TEST 2 IS SELECTED
                    /
0345  0320  KXT1,   XIT1
0346  0341  KXT1C,  XIT1C


                    /
                    /TEST 2,  WRITE CHECKER PATTERN #2
                    /
      0400          *400
                    /
0400  4451  TST2,   JMS I XSETU      /SET DF FOR 1ST FIELD
0401  7200          CLA
0402  1053          TAD K262
0403  3062          DCA TNUM                 /TEST #
0404  7240          CLA CMA
0405  3010          DCA 10           /SET ADDRESS COUNT TO 7777
0406  4463          JMS I XBANK      /SEE IF FIELD HAS PROGRAM
0407  7410          SKP              /NO,  BEGIN WRITING
0410  5223          JMP EXT2                 /DONE ALL,  NOW READ ALL
0411  1250          TAD KXT2
0412  3153          DCA EXIT
0413  4464          JMS I X0011      /WRITE 0011
0414  4465          JMS I X1100      /WRITE 1100 128 TIMES
0415  4465          JMS I X1100
0416  4464          JMS I X0011      /WRITE 0011 128 TIMES
0417  4464          JMS I X0011
```

```
0420  5214          JMP .-4
0421  4471  XIT2,   JMS I XTBNK     /SETUP FOR NEXT FIELD
0422  5204          JMP TST2+4

0423  4500  EXT2,   JMS I XCHK2     /READ EACH FIELD AND CHECK
0424  5225          JMP TST2C               /NOW WRITE COMPLEMENT
                    /
                    /WRITE COMPLEMENT OF PATTERN 2
                    /
0425  4451  TST2C,  JMS I XSETU     /SET DF FOR FIRST FIELD
0426  7240          CLA CMA
0427  3010          DCA 10          /SET ADR, COUNT TO 7777
0430  4463          JMS I XBANK     /SEE IF FIELD HAS PROGRAM
0431  7410          SKP             /WRITE
0432  5245          JMP EXT2C               /GO READ
0433  1251          TAD KXT2C
0434  3153          DCA EXIT
0435  4465          JMS I X1100     /WRITE 1100
0436  4464          JMS I X0011     /WRITE 0011 128 TIMES
0437  4464          JMS I X0011
0440  4465          JMS I X1100
0441  4465          JMS I X1100     /WRITE 1100 128 TIMES
0442  5236          JMP .-4
0443  4471  XIT2C,  JMS I XTBNK     /SETUP FOR NEXT FIELD
0444  5226          JMP TST2C+1
                    /
0445  4501  EXT2C,  JMS I XCHK2C    /READ EACH FIELD AND CHECK
0446  5647          JMP I .+1               /SEE IF TEST 3 IS SELECTED
0447  0253          EXAM3
                    /
0450  0421  KXT2,   XIT2
0451  0443  KXT2C,  XIT2C


                    /
                    /TEST 3.  WRITE CHECKER PATTERN #3
                    /
0452  4451  TST3,   JMS I XSETU     /SETUP FOR 1ST FIELD
0453  7200          CLA
0454  1054          TAD K263
0455  3062          DCA TNUM                /TEST NUMBER
0456  7240          CLA CMA
0457  3010          DCA 10          /SET ADR, COUNT TO 7777
0460  4463          JMS I XBANK     /SEE IF FIELD HAS PROGRAM
0461  7410          SKP             /GO WRITE
0462  5275          JMP EXT3                /GO READ
0463  1322          TAD KXT3
0464  3153          DCA EXIT
0465  4467          JMS I X1001     /WRITE 1001
0466  4466          JMS I X0110     /WRITE 0110 128 TIMES
0467  4466          JMS I X0110
0470  4467          JMS I X1001     /WRITE 1001 128 TIMES
0471  4467          JMS I X1001
0472  5266          JMP .-4
0473  4471  XIT3,   JMS I XTBNK     /SETUP FOR NEXT FIELD
```

```
0474  5256          JMP TST3+4
                    /
0475  4502  EXT3,   JMS I XCHK3     /READ EACH FIELD AND CHECK
0476  5277          JMP TST3C               /WRITE COMPLEMENT
                    /
                    /WRITE COMPLEMENT OF PATTERN 3
                    /
0477  4451  TST3C,  JMS I XSETU     /SETUP DF FOR 1ST FIELD
0500  7240          CLA CMA
0501  3010          DCA 10          /SET ADR, COUNT TO 7777
0502  4463          JMS I XBANK     /SEE IF FIELD HAS PROGRAM
0503  7410          SKP             /WRITE
0504  5317          JMP EXT3C               /READ ALL
0505  1323          TAD KXT3C
0506  3153          DCA EXIT
0507  4466          JMS I X0110     /WRITE 0110
0510  4467          JMS I X1001     /WRITE 1001 128 TIMES
0511  4467          JMS I X1001
0512  4466          JMS I X0110     /WRITE 0110 128 TIMES
0513  4466          JMS I X0110
0514  5310          JMP .-4
0515  4471  XIT3C,  JMS I XTBNK     /SETUP FOR NEXT FIELD
0516  5300          JMP TST3C+1
                    /
0517  4503  EXT3C,  JMS I XCHK3C    /READ EACH FIELD AND CHECK
0520  5721          JMP I .+1               /SEE IF TEST 4 IS SELECTED
0521  0260          EXAM4
                    /
0522  0473  KXT3,   XIT3
0523  0515  KXT3C,  XIT3C


                    /
                    /TEST 4.  WRITE PATTERN #4
                    /
      0600          *600
                    /
0600  4451  TST4,   JMS I XSETU     /SET DF FOR 1ST FIELD
0601  7200          CLA
0602  1055          TAD K264
0603  3062          DCA TNUM                /TEST NUMBER
0604  7240          CLA CMA
0605  3010          DCA 10          /SET ADR, COUNT TO 7777
0606  4463          JMS I XBANK     /SEE IF FIELD HAS PROGRAM
0607  7410          SKP             /WRITE
0610  5223          JMP EXT4                /GO READ
0611  1245          TAD KXT4
0612  3153          DCA EXIT
0613  4466          JMS I X0110     /WRITE 0110
0614  4466          JMS I X0110     /WRITE 0110 64 TIMES
0615  4467          JMS I X1001     /WRITE 1001 128 TIMES
0616  4467          JMS I X1001
0617  4466          JMS I X0110
0620  5214          JMP .-4
0621  4471  XIT4,   JMS I XTBNK     /SETUP FOR NEXT FIELD
```

```
0622  5204          JMP TST4+4
                    /
0623  4504  EXT4,   JMS I XCHK4     /READ EACH FIELD AND CHECK
0624  5225          JMP TST4C               /WRITE COMPLEMENT
                    /
                    /WRITE COMPLEMENT OF PATTERN 4
                    /
0625  4451  TST4C,  JMS I XSETU     /SET DF FOR FIRST
0626  7240          CLA CMA
0627  3010          DCA 10          /SET ADR, COUNT TO 7777
0630  4463          JMS I XBANK     /SEE IF FIELD HAS PROGRAM
0631  7410          SKP             /WRITE
0632  5247          JMP EXT4C               /READ
0633  1246          TAD KXT4C
0634  3153          DCA EXIT
0635  4467          JMS I X1001
0636  4467          JMS I X1001     /WRITE 1001 64 TIMES
0637  4466          JMS I X0110     /WRITE X0110 128 TIMES
0640  4466          JMS I X0110
0641  4467          JMS I X1001
0642  5236          JMP .+4
0643  4471  XIT4C,  JMS I XTBNK     /SETUP FOR NEXT FIELD
0644  5226          JMP TST4C+1
                    /
0645  0621  KXT4,   XIT4
0646  0643  KXT4C,  XIT4C

0647  4505  EXT4C,  JMS I XCHK4C    /READ EACH FIELD AND CHECK
0650  5651          JMP I .+1       /SEE IF READY TO MOVE
0651  0265          EXAM4+5
                    /
                    /ROUTINE TO WRITE 0011
                    /
0652  0000  W0011,  0
0653  1056          TAD M20
0654  3072          DCA COUNT
0655  3410          DCA I 10                /0
0656  3410          DCA I 10                /0
0657  7040          CMA
0660  3410          DCA I 10                /1
0661  7040          CMA
0662  3410          DCA I 10                /1
0663  2072          ISZ COUNT               /COUNT = -16 OR -32
0664  5255          JMP W0011+3     /LOOP
0665  4470          JMS I XKBNK     /SEE IF END OF FIELD
0666  5652          JMP I W0011     /EXIT
                    /
                    /ROUTINE TO WRITE 1100
                    /
0667  0000  W1100,  0
0670  1056          TAD M20
0671  3072          DCA COUNT
0672  7040          CMA
0673  3410          DCA I 10                /1
```

```
0674  7040          CMA
0675  3410          DCA I 10                /1
0676  3410          DCA I 10                /0
0677  3410          DCA I 10                /0
0700  2072          ISZ COUNT               /-16 OR -32
0701  5272          JMP W1100+3     /LOOP
0702  4470          JMS I XKBNK     /SEE IF END OF FIELD
0703  5667          JMP I W1100     /EXIT
                    /
                    /ROUTINE TO WRITE 0110
                    /
0704  0000    W0110,  0
0705  1056          TAD M20
0706  3072          DCA COUNT
0707  3410          DCA I 10                /0
0710  7040          CMA
0711  3410          DCA I 10                /1
0712  7040          CMA
0713  3410          DCA I 10                /1
0714  3410          DCA I 10                /0
0715  2072          ISZ COUNT               /-16 OR -32
0716  5307          JMP W0110+3
0717  4470          JMS I XKBNK     /SEE IIF END OF FIELD
0720  5704          JMP I W0110     /EXIT

                    /
                    /ROUTINE TO WRITE 1001
                    /
0721  0000    W1001,  0
0722  1056          TAD M20
0723  3072          DCA COUNT
0724  7040          CMA
0725  3410          DCA I 10                /1
0726  3410          DCA I 10                /0
0727  3410          DCA I 10                /0
0730  7040          CMA
0731  3410          DCA I 10                /1
0732  2072          ISZ COUNT               /-16 TO -32
0733  5324          JMP W1001+3     /LOOP
0734  4470          JMS I XKBNK     /SEE IF END OF FIELD
0735  5721          JMP I W1001     /EXIT

                    /
                    /ROUTINE TO READ ALL OF MEMORY 8 TIMES, COMPLEMENTING
                    /THE PATTERN EACH PASS.  NO ERROR CHECKING IS DONE.
                    /
0736  0000    RDALL,  0
0737  7200          CLA
0740  1057          TAD M40         /-32 DECIMAL
0741  3072          DCA COUNT               /COUNTS PASSES THRU MEMORY
0742  7240          CLA CMA
0743  3010          DCA 10          /SET ADR, REGS, TO 777
0744  7040          CMA
```

```
0745  3011          DCA 11
0746  3073          DCA FLCNT
0747  4463          JMS I XBANK     /SEE IF FIELD HAS PROGRAM
0750  7410          SKP             /READ
0751  5360          JMP CDON1               /DONE
0752  7200          CLA
0753  1410  RDLOP,  TAD I 10                /READ ONE
0754  7040          CMA
0755  3411          DCA I 11                /WRITE BACK
0756  2073          ISZ FLCNT               /DONE 1 FIELD WHEN SKIP
0757  5353          JMP RDLOP
0760  2072  CDON1,  ISZ COUNT               /DONE 32 PASSES WHEN SKIP
0761  5342          JMP RDALL+4             /DO ANOTHER PASS
0762  5736          JMP I RDALL     /EXIT
                    /
                    /READ AND CHECK FOR ERROR ROUTINE
                    /
      1000          *1000
                    /
1000  0000  RCHKA,  0
1001  4451          JMS I XSETU     /SET DF TO 1ST FIELD
1002  4475          JMS I XRALL     /READ ALL, DON'T CHECK
1003  3122          DCA MEMADR      /SET ADR, COUN  TO 0
1004  4463          JMS I XBANK     /SEE IF FIELD HAS PROGRAM
1005  7410          SKP
1006  5600          JMP I RCHKA
1007  1235          TAD KRXT
1010  3153          DCA EXIT
1011  0000  RLOPA,  0
1012  0000          0               /WILL = JMS I XRD1, 2, 3, OR 4
1013  0000          0
1014  0000          0
1015  0000          0
1016  7000          NOP             /WILL = NOP OR JMS STALL
1017  5212          JMP RLOPA+1
1020  1216  RXIT,   TAD .-2
1021  1236          TAD K1K
1022  7640          SZA CLA         /NOP IF 0
1023  5227          JMP .+4
1024  1114          TAD JMS5        /=JMS I XSALL
1025  3216          DCA RXIT-2
1026  5203          JMP RCHKA+3     /READ SLOW
1027  7000          NOP
1030  1227          TAD .-1
1031  3216          DCA RXIT-2
1032  4471          JMS I XTBNK     /SETUP FOR NEXT FIELD
1033  5202          JMP RCHKA+2
1034  5600          JMP I RCHKA     /EXIT
                    /
1035  1020  KRXT,   RXIT
1036  1000  K1K,    1000


                    /
                    /SETUP ROUTINES FOR RCHKA
```

```
                        /
1037  0000  RCHK1,  0
1040  1110         TAD JMS1          /JMS1 = JMS I XRD1
1041  3211         DCA RLOPA
1042  1110         TAD JMS1                  /JMS1 = JMS I XRD1
1043  3212         DCA RLOPA+1
1044  1111         TAD JMS2
1045  3213         DCA RLOPA+2
1046  1111         TAD JMS2
1047  3214         DCA RLOPA+3
1050  1110         TAD JMS1
1051  3215         DCA RLOPA+4
1052  4200         JMS RCHKA                 /GO READ
1053  5637         JMP I RCHK1       /EXIT
                        /
1054  0000  RCHK1C, 0
1055  1111         TAD JMS2          /JMS2 = JMS I XRD2
1056  3211         DCA RLOPA
1057  1111         TAD JMS2
1060  3212         DCA RLOPA+1
1061  1110         TAD JMS1
1062  3213         DCA RLOPA+2
1063  1110         TAD JMS1
1064  3214         DCA RLOPA+3
1065  1111         TAD JMS2
1066  3215         DCA RLOPA+4
1067  4200         JMS RCHKA                 /GO READ
1070  5654         JMP I RCHK1C      /EXIT
                        /
1071  0000  RCHK2,  0
1072  1110         TAD JMS1          /JMS1 = JMS I XRD1
1073  3211         DCA RLOPA
1074  1111         TAD JMS2                  /JMS I XRD2
1075  3212         DCA RLOPA+1
1076  1111         TAD JMS2
1077  3213         DCA RLOPA+2
1100  1110         TAD JMS1                  /JMS I XRD1
1101  3214         DCA RLOPA+3
1102  1110         TAD JMS1
1103  3215         DCA RLOPA+4
1104  4200         JMS RCHKA                 /GO READ
1105  5671         JMP I RCHK2       /EXIT
                        /
1106  0000  RCHK2C, 0
1107  1111         TAD JMS2
1110  3211         DCA RLOPA
1111  1110         TAD JMS1          /JMS I XRD1
1112  3212         DCA RLOPA+1
1113  1110         TAD JMS1
1114  3213         DCA RLOPA+2
1115  1111         TAD JMS2
1116  3214         DCA RLOPA+3
1117  1111         TAD JMS2
1120  3215         DCA RLOPA+4
1121  4200         JMS RCHKA                 /GO READ
```

```
1122  5706            JMP I RCHK2C    /EXIT


                      /
1123  0000    RCHK3,  0
1124  1113            TAD JMS4        /JMS I XRD4
1125  3211            DCA RLOPA
1126  1112            TAD JMS3                /JMS I XRD3
1127  3212            DCA RLOPA+1
1130  1112            TAD JMS3        /JMS I XRD3
1131  3213            DCA RLOPA+2
1132  1113            TAD JMS4                /JMS I XRD4
1133  3214            DCA RLOPA+3
1134  1113            TAD JMS4
1135  3215            DCA RLOPA+4
1136  4200            JMS RCHKA               /GO READ
1137  5723            JMP I RCHK3     /EXIT
                      /
1140  0000    RCHK3C, 0
1141  1112            TAD JMS3        /JMS I XRD3
1142  3211            DCA RLOPA
1143  1113            TAD JMS4                /JMS I XRD4
1144  3212            DCA RLOPA+1
1145  1113            TAD JMS4
1146  3213            DCA RLOPA+2
1147  1112            TAD JMS3
1150  3214            DCA RLOPA+3
1151  1112            TAD JMS3
1152  3215            DCA RLOPA+4
1153  4200            JMS RCHKA               /GO READ
1154  5740            JMP I RCHK3C    /EXIT
                      /
      1200    *1200
                      /
1200  0000    RCHK4,  0
1201  4531            JMS I XFILD
1202  1112            TAD JMS3        /JMS I XRD3
1203  3637            DCA I XLOPA
1204  1112            TAD JMS3                /JMS I XRD3
1205  3640            DCA I XLOPB
1206  1113            TAD JMS4
1207  3641            DCA I XLOPC
1210  1113            TAD JMS4                /JMS I XRD4
1211  3642            DCA I XLOPD
1212  1112            TAD JMS3
1213  3643            DCA I XLOPE
1214  4644            JMS I XCFL
1215  4636            JMS I XCHKA             /GO READ
1216  5600            JMP I RCHK4     /EXIT
                      /
1217  0000    RCHK4C, 0
1220  4531            JMS I XFILD
1221  1113            TAD JMS4        /JMS I XRD4
1222  3637            DCA I XLOPA
1223  1113            TAD JMS4
```

```
1224  3640            DCA I XLOPB
1225  1112            TAD JMS3
1226  3641            DCA I XLOPC
1227  1112            TAD JMS3
1230  3642            DCA I XLOPD
1231  1113            TAD JMS4
1232  3643            DCA I XLOPE
1233  4644            JMS I XCFL
1234  4636            JMS I XCHKA             /GO READ
1235  5617            JMP I RCHK4C    /EXIT

1236  1000    XCHKA,  RCHKA
1237  1011    XLOPA,  RLOPA
1240  1012    XLOPB,  RLOPA+1
1241  1013    XLOPC,  RLOPA+2
1242  1014    XLOPD,  RLOPA+3
1243  1015    XLOPE,  RLOPA+4
1244  1722    XCFL,   CFLD

                PAUSE


                /81-8S EXTENDED CHECKERBOARD - TAPE 2
                /
                /READ ROUTINES FOR 0011; 1100; 0110 AND 1001
                /
1245  0000    RD1,    0
1246  1056            TAD M20         /-16
1247  3072            DCA COUNT
1250  1060            TAD M4          /-4
1251  3073            DCA FLCNT
1252  1141    CLOP1,  TAD M10         /-8
1253  3074            DCA LOOP
1254  1522            TAD I MEMADR
1255  7040            CMA
1256  3522            DCA I MEMADR
1257  2074            ISZ LOOP                /COMPLEMENT 8 TIMES
1260  5254            JMP .-4
1261  2073            ISZ FLCNT               /DONE 4 ADRS, WHEN SKIP
1262  7410            SKP
1263  5266            JMP .+3
1264  2122            ISZ MEMADR
1265  5252            JMP CLOP1
                /
1266  1122            TAD MEMADR
1267  1060            TAD M4          /SUBTRACT 4
1270  3010            DCA 10          /NOW USE AUTO-INDEX
1271  7100            CLL
1272  1410            TAD I 10
1273  7440            SZA             /0
1274  4521            JMS I XRROR     /PRINT ERROR
1275  7100            CLL
1276  1410            TAD I 10
1277  7440            SZA             /0
1300  4521            JMS I XRROR     /PRINT ERROR
```

```
1301  7120          STL
1302  1410          TAD I 10
1303  7040          CMA             /1
1304  7440          SZA
1305  4521          JMS I XRROR     /PRINT ERROR
1306  7120          STL
1307  1410          TAD I 10
1310  7040          CMA             /1
1311  7440          SZA
1312  4521          JMS I XRROR     /PRINT ERROR
1313  2072          ISZ COUNT
1314  5320          JMP .+4
1315  4470          JMS I XKBNK             /SEE IF END OF FIELD
1316  2122          ISZ MEMADR
1317  5645          JMP I RD1
                /
1320  2122          ISZ MEMADR      /KEEP READING
1321  5250          JMP RD1+3


                /
1322  0000  RD2,     0
1323  1056          TAD M20         /-16
1324  3072          DCA COUNT
1325  1060          TAD M4          /-4
1326  3073          DCA FLCNT

1327  1141  CLOP2,  TAD M10         /-8
1330  3074          DCA LOOP
1331  1522          TAD I MEMADR    /READ
1332  7040          CMA
1333  3522          DCA I MEMADR
1334  2074          ISZ LOOP                /COMPLEMENT 4 TIMES
1335  5331          JMP .-4
1336  2073          ISZ FLCNT               /DONE 4 ADRS', WHEN SKIP
1337  7410          SKP
1340  5343          JMP .+3
1341  2122          ISZ MEMADR      /INCREMENT ADDRESS
1342  5327          JMP CLOP2
                /
1343  1122          TAD MEMADR
1344  1060          TAD M4
1345  3010          DCA 10          /NOW USE AUTO-INDEX
1346  7120          STL
1347  1410          TAD I 10
1350  7040          CMA             /1
1351  7440          SZA
1352  4521          JMS I XRROR     /PRINT ERROR
1353  7120          STL
1354  1410          TAD I 10
1355  7040          CMA             /1
1356  7440          SZA
1357  4521          JMS I XRROR     /PRINT ERROR
1360  7100          CLL
1361  1410          TAD I 10
```

```
1362  7440          SZA                 /0
1363  4521          JMS I XRROR         /PRINT ERROR
1364  7100          CLL
1365  1410          TAD I 10
1366  7440          SZA                 /0
1367  4521          JMS I XRROR         /PRINT ERROR
1370  2072          ISZ COUNT
1371  5375          JMP .+4
1372  4470          JMS I XKBNK                 /SEE IF END OF FIELD
1373  2122          ISZ MEMADR
1374  5722          JMP I RD2
                /
1375  2122          ISZ MEMADR                  /KEEP READING
1376  5325          JMP RD2+3


                /
      1400    *1400
                /
1400  0000    RD3,  0
1401  1056          TAD M20                     /-16
1402  3072          DCA COUNT
1403  1060          TAD M4              /-4
1404  3073          DCA FLCNT
1405  1141    CLOP3,TAD M10             /-8
1406  3074          DCA LOOP
1407  1522          TAD I MEMADR        /READ
1410  7040          CMA
1411  3522          DCA I MEMADR
1412  2074          ISZ LOOP                    /COMPLEMENT 8 TIMES
1413  5207          JMP .-4
1414  2073          ISZ FLCNT                   /DONE 4 IF 0
1415  7410          SKP
1416  5221          JMP .+3
1417  2122          ISZ MEMADR
1420  5205          JMP CLOP3                   /DO NEXT
                /
1421  1122          TAD MEMADR
1422  1060          TAD M4
1423  3010          DCA 10              /USE AUTO-INDEX
1424  7100          CLL
1425  1410          TAD I 10
1426  7440          SZA                 /0
1427  4521          JMS I XRROR         /PRINT ERROR
1430  7120          STL
1431  1410          TAD I 10
1432  7040          CMA                 /1
1433  7440          SZA
1434  4521          JMS I XRROR         /PRINT ERROR
1435  7120          STL
1436  1410          TAD I 10
1437  7040          CMA                 /1
1440  7440          SZA
1441  4521          JMS I XRROR         /PRINT ERROR
1442  7100          CLL
```

```
1443  1410            TAD I 10
1444  7440            SZA             /0
1445  4521            JMS I XRROR     /PRINT ERROR
1446  2072            ISZ COUNT
1447  5253            JMP .+4
1450  4470            JMS I XKBNK     /SEE IF END OF FIELD
1451  2122            ISZ MEMADR
1452  5600            JMP I RD3
                /
1453  2122            ISZ MEMADR
1454  5203            JMP RD3+3


                /
1455  0000    RD4,    0
1456  1056            TAD M20         /-16
1457  3072            DCA COUNT
1460  1060            TAD M4          /-4
1461  3073            DCA FLCNT
1462  1141    CLOP4,  TAD M10         /-8
1463  3074            DCA LOOP
1464  1522            TAD I MEMADR
1465  7040            CMA
1466  3522            DCA I MEMADR
1467  2074            ISZ LOOP                /COMPLEMENT 8 TIMES
1470  5264            JMP .-4
1471  2073            ISZ FLCNT               /DONE 4 ADRS', WHEN SKIP
1472  7410            SKP
1473  5276            JMP .+3
1474  2122            ISZ MEMADR      /INCREMENT ADDRESS
1475  5262            JMP CLOP4
1476  1122            TAD MEMADR
1477  1060            TAD M4
1500  3010            DCA 10          /USE AUTO-INDEX
1501  7120            STL
1502  1410            TAD I 10
1503  7040            CMA             /1
1504  7440            SZA
1505  4521            JMS I XRROR     /PRINT ERROR
1506  7100            CLL
1507  1410            TAD I 10
1510  7440            SZA             /0
1511  4521            JMS I XRROR     /PRINT ERROR
1512  7100            CLL
1513  1410            TAD I 10
1514  7440            SZA             /0
1515  4521            JMS I XRROR     /PRINT ERROR
1516  7120            STL
1517  1410            TAD I 10
1520  7040            CMA             /1
1521  7440            SZA
1522  4521            JMS I XRROR     /PRINT ERROR
1523  2072            ISZ COUNT
1524  5330            JMP .+4
1525  4470            JMS I XKBNK             /SEE IF END OF FIELD
```

```
1526  2122            ISZ MEMADR
1527  5655            JMP I RD4
                      /
1530  2122            ISZ MEMADR
1531  5260            JMP RD4+3


                      /
                      /ROUTINE TO CHECK FOR END OF FIELD
                      /
      1600            *1600
1600  0000      CKBNK, 0
1601  7200            CLA
1602  1010            TAD 10
1603  7040            CMA
1604  7640            SZA CLA
1605  5600            JMP I CKBNK     /NOT DONE
1606  5553            JMP I EXIT      /DONE
                      /
                      /ROUTINE TO SEE IF TESTED FIELD HAS PROGRAM
                      /
1607  0000      CBANK, 0
1610  6224            RIF             /READ INST, FIELD
1611  3223            DCA SAVIF               /SAVE
1612  6214            RDF             /READ DATA FIELD
1613  7041            CIA
1614  1223            TAD SAVIF
1615  7640            SZA CLA         /EQUAL IF AC=0
1616  5607            JMP I CBANK     /DOESN'T HAVE PROGRAM
1617  4471            JMS I XTBNK     /INCREMENT DATA FIELD
1620  5607            JMP I CBANK     /TEST NEW FIELD
1621  2207            ISZ CBANK               /DONE ALL CAUSE PROGRAM NOW
                                      /IN HIGHEST FIELD
1622  5607            JMP I CBANK     /EXIT
1623  0000      SAVIF, 0


                      /
                      /ROUTINE TO SET DF FOR NEXT FIELD
                      /
1624  0000      NXTBNK, 0
1625  7200            CLA
1626  6214            RDF             /READ DATA FIELD
1627  7041            CIA
1630  1124            TAD LAST1               /C(LAST1) = LAST TO TEST
1631  7640            SZA CLA         /ALL DONE IF 0
1632  5235            JMP .+3
1633  2224            ISZ NXTBNK
1634  5242            JMP .+6                 /EXIT
1635  6214            RDF
1636  1034            TAD K10         /INCREMENT DATA FIELD
1637  1125            TAD KCDF                /ADD .6201
1640  3241            DCA .+1
1641  6201            CDF 00          /CHANGE TO NEW DATA FIELD
                      /
```

```
                        /CHECK SWITCH REGISTER
1642  7604              LAS
1643  7712              SPA CLA           /CHEC HALT
1644  4527              JMS I XHLT        /GO HALT, SR0=1
1645  5624              JMP I NXTBNK      /EXIT
                        /
                        /RESTORE DATA FIELD AND CHECK SR
                        /
1646  0000    FEILD,    0
1647  7200              CLA
1650  6214              RDF
1651  3014              DCA DATFLD        /SAVE TESTED FIELD#
1652  6224              RIF
1653  1125              TAD KCDF
1654  3255              DCA .+1
1655  6201              CDF 00            /MAKE DATA AND INST FIELD EQUAL
1656  7200              CLA
1657  5646              JMP I FEILD


                        /
      1700              *1700
                        /
                        /START HERE TO LOOP ON ADDRESS
                        /
1700  7200              CLA
1701  7604              LAS               /READ LOWER LIMIT
1702  3123              DCA FIRST1
1703  7402              HLT               /NOW SETUP UPPER LIMIT
1704  7604              LAS
1705  3124              DCA LAST1
1706  1123    OVER,     TAD FIRST1
1707  3122              DCA MEMADR
1710  1522    WRLOP,    TAD I MEMADR      /READ
1711  3522              DCA I MEMADR      /WRITE
1712  1122              TAD MEMADR
1713  7041              CIA
1714  1124              TAD LAST1
1715  7650              SNA CLA
1716  5306              JMP OVER
1717  2122              ISZ MEMADR
1720  5310              JMP WRLOP
1721  7402              HLT
                        /
1722  0000    CFLD,     0
1723  7200              CLA
1724  1014              TAD DATFLD        /TEST FIELD
1725  1125              TAD KCDF
1726  3327              DCA .+1
1727  6201              CDF 00            /RESTORE TEST FIELD
1730  7200              CLA
1731  5722              JMP I CFLD        /EXIT


                        /
```

```
                    /PRINT ERROR ROUTINE
                    /
          2000      *2000
  2000    0000      ERROR,  0
  2001    7430              SZL             /READING 1'S IF LINK = 1
  2002    7040              CMA
  2003    3363              DCA BAD         /SAVE BAD DATA
  2004    7430              SZL
  2005    7040              CMA
  2006    3364              DCA GOOD                /SAVE GOOD DATA
  2007    1010              TAD 10
  2010    3365              DCA OCADR               /OCTAL ADDRESS
  2011    4531              JMS I XFILD     /RESTORE DATA FIELD
  2012    1014              TAD DATFLD      /DATA FIELD
  2013    7041              CIA
  2014    1366              TAD LAST                /LAST = FIELD WITH LAST ERROR
  2015    7650              SNA CLA         /SAME IF 0
  2016    5233              JMP SW2         /DON'T STORE
  2017    1014              TAD DATFLD
  2020    3366              DCA LAST
  2021    1022              TAD ERWRD               /TABLE POINTER
  2022    7041              CIA
  2023    1021              TAD ENTBL
  2024    7640              SZA CLA         /END OF TABLE IF = 0
  2025    5230              JMP .+3
  2026    1020              TAD ERTBL
  2027    3022              DCA ERWRD               /RESTORE POINTER
  2030    1014              TAD DATFLD
  2031    2022              ISZ ERWRD               /INCREMENT POINTER
  2032    3422              DCA I ERWRD     /STORE IN TABLE
                    /
  2033    7604      SW2,    LAS
  2034    7006              RTL
  2035    7700              SMA CLA         /SR2 ON A 1 = RING BELL
  2036    5242              JMP SW1
  2037    1367              TAD K207
  2040    4346              JMS PRERR               /RING BELL
  2041    5311              JMP SW0
  2042    7604      SW1,    LAS
  2043    7004              RAL
  2044    7700              SMA CLA         /SR1 A 1 = NO PRINT
  2045    5254              JMP EPRNT
  2046    1014      EREXT,  TAD DATFLD
  2047    1125              TAD KCDF
  2050    3251              DCA .+1
  2051    6201              CDF 00          /SET TO TESTED FIELD
  2052    7200              CLA
  2053    5600              JMP I ERROR


                    /
  2054    4354      EPRNT,  JMS CRLF                /CR,LF
  2055    1014              TAD DATFLD              /TEST NUMBER
  2056    7012              RTR
  2057    7010              RAR
```

```
2060  1142          TAD K260
2061  4346          JMS PRERR               /PRINT
2062  1142          TAD M14           /-12 DECIMAL
2063  3074          DCA LOOP
2064  4315          JMS SPING               /SPACE 12
2065  1365          TAD OCADR               /OCTAL ADR'
2066  3134          DCA CHAR                /SAVE
2067  4326          JMS PROCTL        /PRINT
2070  1141          TAD M10           /-8 DECIMAL
2071  3074          DCA LOOP
2072  4315          JMS SPING               /SPACE  8
2073  1364          TAD GOOD
2074  3134          DCA CHAR
2075  4326          JMS PROCTL        /PRINT
2076  1061          TAD M5            /-5
2077  3074          DCA LOOP
2100  4315          JMS SPING               /SPACE 5
2101  1363          TAD BAD
2102  3134          DCA CHAR
2103  4326          JMS PROCTL        /PRINT
2104  1061          TAD M5
2105  3074          DCA LOOP
2106  4315          JMS SPING               /SPACE 5
2107  1062          TAD TNUM                /TEST NUMBER
2110  4346          JMS PRERR               /PRINT

                    /
2111  7604    SW0,  LAS
2112  7710          SPA CLA           /CH CK SR0
2113  4527          JMS I XHLT              /GO HALT
2114  5246          JMP EREXT               /EXIT
                    /
                    /PRINT SPACES
                    /
2115  0000    SPING, 0
2116  1370          TAD K240                /SPACE
2117  6046          TLS               /PRINT
2120  6041          TSF
2121  5320          JMP .-1
2122  2074          ISZ LOOP
2123  5317          JMP SPING+2
2124  7200          CLA
2125  5715          JMP I SPING       /EXIT


                    /
                    /PRINT OCTAL
                    /
2126  0000    PROCTL, 0
2127  1060          TAD M4            /-4
2130  3074          DCA LOOP                /DIGIT COUNTER
2131  1134    POSITN, TAD CHAR
2132  7104          CLL RAL
2133  7006          RTL
2134  3134          DCA CHAR
```

```
2135  1134          TAD CHAR
2136  7004          RAL
2137  0133          AND K7
2140  1142          TAD K260                /MAKE ASCII
2141  4346          JMS PRERR               /PRINT ONE
2142  2074          ISZ LOOP
2143  5331          JMP POSITN      /DO NEXT
2144  7200          CLA
2145  5726          JMP I PROCTL    /EXIT
                    /
                    /PRINT A NUMBER
                    /
2146  0000    PRERR, 0
2147  6046          TLS
2150  6041          TSF
2151  5350          JMP .-1
2152  7200          CLA
2153  5746          JMP I PRERR     /EXIT
                    /
                    /CARRIAGE RETURN, LINE FEED
                    /
2154  0000    CRLF,  0
2155  7200          CLA
2156  1143          TAD K215                /CR
2157  4346          JMS PRERR
2160  1371          TAD K212                /LF
2161  4346          JMS PRERR
2162  5754          JMP I CRLF
                    /
2163  0000    BAD,   0
2164  0000    GOOD,  0
2165  0000    OCADR, 0
2166  0000    LAST,  0
2167  0207    K207,  207
2170  0240    K240,  240
2171  0212    K212,  212


                    /
      2200          *2200
                    /
                    /ROUTINE TO SET DF TO FIRST TEST FIELD
                    /
2200  0000    SETU1, 0
2201  7200          CLA
2202  1123          TAD FIRST1      /FIRST TO TEST
2203  1125          TAD KCDF
2204  3205          DCA .+1

2205  6201          CDF 00          /CHANGE TO TEST FIELD
2206  5600          JMP I SETU1     /EXIT
                    /
                    /ROUTINE TO ACCEPT TEST LIMITS FROM
                    /KEYBOARD INPUT
                    /
```

```
2207  0000  SLMTS,  2
2210  4531          JMS I XFILD
2211  4547          JMS I XCRLF     /CR, LF
2212  4751          JMS I XTLIM     /PRINT TEST LIMITS
2213  4547          JMS I XCRLF     /CR, LF
2214  4303          JMS KEYIN               /GO ACCEPT INPUT
2215  4314          JMS LEGAL               /SEE IF IT'S LEGAL
2216  1134          TAD CHAR
2217  0133          AND K7          /MASK AC 9-11
2220  7104          CLL RAL
2221  7006          RTL             /POSITION TO AC 6-8
2222  3123          DCA FIRST1      /FIRST TO TEST
2223  4303          JMS KEYIN               /WAIT FOR COMMA
2224  1134          TAD CHAR                /GET INPUT
2225  7041          CIA
2226  1353          TAD K254
2227  7450          SNA             /OK IF 0
2230  5233          JMP .+3
2231  4344          JMS QUERY               /PRINT QUESTION MARK
2232  5211          JMP SLMTS+2
2233  4303          JMS KEYIN               /WAIT FOR 2ND
2234  4314          JMS LEGAL               /SEE IF IT'S LEGAL
2235  1134          TAD CHAR
2236  0133          AND K7          /MASK AC 9-11
2237  7104          CLL RAL
2240  7006          RTL             /POSITION TO AC 6-8
2241  3124          DCA LAST1               /LAST TO TEST
2242  1123          TAD FIRST1
2243  7041          CIA
2244  1124          TAD LAST1
2245  7500          SMA             /1ST IS > LAST IF NEG
2246  5256          JMP OKAS
2247  7200          CLA


2250  1123          TAD FIRST1
2251  3134          DCA CHAR
2252  1124          TAD LAST1
2253  3123          DCA FIRST1      /LAST NOW IS FIRST
2254  1134          TAD CHAR
2255  3124          DCA LAST1               /FIRST IS NOW LAST
2256  7200  OKAS,   CLA
2257  1124          TAD LAST1
2260  7041          CIA
2261  1123          TAD FIRST1      /SEE IF EQUAL
2262  7440          SZA             /YES IF 0
2263  5273          JMP ALOK
2264  1123          TAD FIRST1      /NOW SEE IF IT HAS PROGRAM
2265  7041          CIA
2266  1016          TAD INSFLD      /CURRENT FIELD
2267  7640          SZA CLA         /NO IF A 1
2270  5273          JMP ALOK
2271  4752          JMS I XLCAT             /PRINT PROGRAM LOCATION
2272  5211          JMP SLMTS+2     /AND START OVER
2273  4303  ALOK,   JMS KEYIN               /WAIT FOR C.R.
```

```
2274  1134          TAD CHAR
2275  7041          CIA
2276  1143          TAD K215
2277  7450          SNA                 /NOT A C.R, IF A SKIP
2300  5607          JMP I SLMTS
2301  4344          JMS QUERY                   /PRINT QUESTION MARK
2302  5211          JMP SLMTS+2         /START OVER
                    /
2303  0000    KEYIN, 0
2304  6032          KCC
2305  6031          KSF
2306  5305          JMP .-1
2307  6036          KRB
2310  3134          DCA CHAR
2311  1134          TAD CHAR
2312  4536          JMS I XPERR
2313  5703          JMP I KEYIN


                    /
2314  0000    LEGAL, 0
2315  1134          TAD CHAR
2316  7041          CIA
2317  1144          TAD K377
2320  7650          SNA CLA             /RUB-OUT IF 0
2321  5211          JMP SLMTS+2
2322  1134          TAD CHAR
2323  0145          AND K370
2324  7041          CIA
2325  1142          TAD K260
2326  7650          SNA CLA
2327  5714          JMP I LEGAL
2330  1134          TAD CHAR
2331  7041          CIA
2332  1353          TAD K254
2333  7650          SNA CLA             /A COMMA IF 0
2334  5714          JMP I LEGAL
2335  1134          TAD CHAR
2336  7041          CIA
2337  1143          TAD K215
2340  7650          SNA CLA             /A C.R, IF 0
2341  5714          JMP I LEGAL
2342  4344          JMS QUERY                   /QUERY
2343  5211          JMP SLMTS+2         /START OVER
                    /
2344  0000    QUERY, 0
2345  4547          JMS I XCRLF
2346  1146          TAD K277
2347  4536          JMS I XPERR         /PRINT QUERY MARK
2350  5744          JMP I QUERY
                    /
2351  2446    XTLIM, TLIMT
2352  2400    XLCAT, LOCAT
2353  0254    K254,  254
```

```
                    /
                    /PRINT FIELD PROGRAM IS IN
                    /
          2400      *2400
                    /
    2400  0000      LOCAT,  0
    2401  7200              CLA
    2402  1016              TAD INSFLD      /CURRENT FIELD
    2403  7012              RTR
    2404  7010              RAR
    2405  0133              AND K7
    2406  1142              TAD K260
    2407  3244              DCA FLDN                /FLDN = PRGAM+25
    2410  1217              TAD PRGAM
    2411  3012              DCA 12
    2412  1412      PLOCT,  TAD I 12
    2413  7450              SNA             /DONE IF 0
    2414  5600              JMP I LOCAT     /EXIT
    2415  4536              JMS I XPERR     /PRINT
    2416  5212              JMP PLOCT
                    /
    2417  2417      PRGAM,  .
    2420  0320              320             /P
    2421  0322              322             /R
    2422  0317              317             /O
    2423  0307              307             /G
    2424  0322              322             /R
    2425  0301              301             /A
    2426  0315              315             /M
    2427  0240              240             /
    2430  0311              311             /I
    2431  0323              323             /S
    2432  0240              240
    2433  0311              311             /I
    2434  0316              316             /N
    2435  0240              240             /
    2436  0306              306             /F
    2437  0311              311             /I
    2440  0305              305             /E
    2441  0314              314             /L
    2442  0304              304             /D
    2443  0240              240             /
    2444  0000      FLDN,   0               /X
    2445  0000              0               /TERMINATOR


                    /
    2446  0000      TLIMT,  0               /PRINT TEST LIMITS
    2447  7200              CLA
    2450  1257              TAD TSTL
    2451  3012              DCA 12
    2452  1412      PLIMT,  TAD I 12
    2453  7450              SNA             /DONE IF 0
    2454  5646              JMP I TLIMT             /DONE IF 0
```

```
2455  4536          JMS I XPERR
2456  5252          JMP PLIMT
                    /
2457  2457    TSTL, .
2460  0324          324             /T
2461  0305          305             /E
2462  0323          323             /S
2463  0324          324             /T
2464  0240          240             /
2465  0314          314             /L
2466  0311          311             /I
2467  0315          315             /M
2470  0311          311             /I
2471  0324          324             /T
2472  0323          323             /S
2473  0000          0               /TERMINATOR
                    /
                    /HEADER ROUTINE
                    /
2474  0000    PHDR, 0
2475  4547          JMS I XCRLF     /CR, LF
2476  1332          TAD FILD
2477  3012          DCA 12
2500  1412    PFILD, TAD I 12                 /PRINT FIELD
2501  7450          SNA             /DONE IF 0
2502  5305          JMP .+3
2503  4536          JMS I XPERR
2504  5300          JMP PFILD
2505  1061          TAD M5
2506  3074          DCA LOOP
2507  4537          JMS I XPING     /SPACE 5
2510  1341          TAD OTLDR
2511  3012          DCA 12

2512  1412    POCDR, TAD I 12                 /PRINT OCTAL ADR
2513  7450          SNA             /DONE IF 0
2514  5317          JMP .+3
2515  4536          JMS I XPERR
2516  5312          JMP POCDR
                    /
2517  1061          TAD M5
2520  3074          DCA LOOP
2521  4537          JMS I XPING     /SPACE 5
2522  1355          TAD GODD
2523  3012          DCA 12
2524  1412    PGOOD, TAD I 12                 /PRINT GOOD
2525  7450          SNA             /DONE IF 0
2526  5731          JMP I .+3
2527  4536          JMS I XPERR
2530  5324          JMP PGOOD
2531  2600          BSPCE           /NEXT PAGE
                    /
2532  2532    FILD, .
2533  0306          306             /F
```

```
2534  0311            311             /I
2535  0305            305             /E
2536  0314            314             /L
2537  0304            304             /D
2540  0000            0
                /
2541  2541    OTLDR,  .
2542  0317            317             /O
2543  0303            303             /C
2544  0324            324             /T
2545  0301            301             /A
2546  0314            314             /L
2547  0240            240
2550  0301            301             /A
2551  0304            304             /D
2552  0322            322             /R
2553  0256            256                     /.
2554  0000            0
                /
2555  2555    GOOD,   .
2556  0307            307             /G
2557  0317            317             /O
2560  0317            317             /O
2561  0304            304             /D
2562  0000            0
                /
2563  5674    EXHDR,  JMP I PHDR


                /
      2600      *2600
                /
2600  1061    BSPCE,  TAD M5
2601  3074            DCA LOOP
2602  4537            JMS I XPING     /SPACE 5
2603  1234            TAD BADD
2604  3012            DCA 12
2605  1412    PBAD,   TAD I 12                /PRINT BAD
2606  7450            SNA             /DONE IF 0
2607  5212            JMP .+3
2610  4536            JMS I XPERR
2611  5205            JMP PBAD
2612  1061            TAD M5
2613  3074            DCA LOOP
2614  4537            JMS I XPING             /SPACE 5
                /
2615  1226            TAD TSTN
2616  3012            DCA 12
2617  1412    PTSTN,  TAD I 12                /PRINT TEST
2620  7450            SNA                     /DONE IF 0
2621  5224            JMP .+3
2622  4536            JMS I XPERR
2623  5217            JMP PTSTN
2624  4547            JMS I XCRLF             /CR, LF
2625  5644            JMP I XPHDR             /EXIT
```

```
                    /
2626  2626  TSTN,   .
2627  0324          324             /T
2630  0305          305             /E
2631  0323          323             /S
2632  0324          324             /T
2633  0000          0
                    /
2634  2634  BADD,   .
2635  0302          302             /B
2636  0301          301             /A
2637  0304          304             /D
2640  0000          0
                    /
2641  0000  HALT,   0
2642  7402          HLT
2643  5641          JMP I HALT      /RESTART HERE OR RTRN1
                    /
2644  2563  XPHDR,  EXHDR


                    /
                    /
                    /WAIT HERE TO SETUP SR, TYPE CARRIAGE RETURN
                    /AFTER SETTING SR,
                    /
2645  0000  SETSR,  0
2646  4531          JMS I XFILD     /RESTORE DATA FIELD
2647  4547          JMS I XCRLF     /CR, LF
2650  1270          TAD STSR
2651  3012          DCA 12
2652  1412  PSTSR,  TAD I 12                /PRINT SETUP SR
2653  7450          SNA             /DONE IF 0
2654  5257          JMP .+3
2655  4536          JMS I XPERR
2656  5252          JMP PSTSR
2657  6036          KRB
2660  6031  WTCR,   KSF
2661  5265          JMP .+4
2662  6036          KRB
2663  4536          JMS I XPERR
2664  5645          JMP I SETSR
2665  7604          LAS
2666  3033          DCA MCWA
2667  5260          JMP WTCR
                    /
2670  2670  STSR,   .
2671  0323          323             /S
2672  0305          305             /E
2673  0324          324             /T
2674  0325          325             /U
2675  0320          320             /P
2676  0240          240
2677  0323          323             /S
```

```
2700  0322            322             /R
2701  0000            0
                      /
2702  0000      STALL,  0
2703  4531              JMS I XFILD
2704  4316              JMS GENRAN      /GET ANOTHER
2705  3074              DCA LOOP
2706  2074              ISZ LOOP        /18.5 MS MAX,
2707  5306              JMP .-1
2710  1014              TAD DATFLD
2711  1125              TAD KCDF
2712  3313              DCA .+1
2713  6201              CDF 00          /RESTORE DATA FIELD
2714  7200              CLA
2715  5702              JMP I STALL     /EXIT

2716  0000      GENRAN, 0
2717  1354              TAD RANTAB
2720  7041              CIA
2721  1342              TAD RANDEX
2722  7640              SZA CLA
2723  5333              JMP RANTAD-1
2724  1355              TAD TBLRAN
2725  3342              DCA RANDEX
2726  1341              TAD RANCON
2727  7104              CLL RAL
2730  7430              SZL
2731  1150              TAD K1
2732  3341              DCA RANCON
2733  1742              TAD I RANDEX
2734  1341      RANTAD, TAD RANCON
2735  3742              DCA I RANDEX
2736  1742              TAD I RANDEX
2737  2342              ISZ RANDEX
2740  5716              JMP I GENRAN
                      /
2741  1234      RANCON, 1234
2742  2753      RANDEX, RANTBL+10
2743  4321      RANTBL, 4321
2744  1416              1416
2745  5363              5363
2746  6060              6060
2747  3035              3035
2750  2572              2572
2751  3237              3237
2752  0214              0214
2753  0000              0
2754  2753      RANTAB, .-1
2755  2743      TBLRAN, RANTBL
2756  0177      K177,   177


                      /
                      /ROUTINE TO DETERMINE FIELD FOR RELOCATION
```

```
                    /
          3000      *3000
                    /
  3000    4531      CMOVE,  JMS I XFILD       /SET DF TO CURRENT FIELD
  3001    7600              7600
  3002    1020              TAD ERTBL
  3003    3022              DCA ERWRD               /SETUP ERROR TABLE POINTER
  3004    1123              TAD FIRST1              /FIRST TESTED FIELD
  3005    7041              CIA
  3006    1124              TAD LAST1               /LAST TESTED FIELD
  3007    7650              SNA CLA           /DON'T MOVE IF EQUAL
  3010    5530              JMP I XRTN              /START OVER
  3011    1015              TAD FLAGS
  3012    7010              RAR
  3013    7430              SZL     /FIRST MOVE IF A SKIP
  3014    5725              JMP I XTMV              /SETUP FOR NEXT MOVE
  3015    7001              IAC                     /SET BIT 11
  3016    3015              DCA FLAGS

  3017    1124              TAD LAST1               /LAST TO TEST = 1ST MOVE
  3020    3016              DCA INSFLD        /NEW CURRENT FIELD
  3021    1016              TAD INSFLD
  3022    1141              TAD M10           /SUBTRACT 1 FROM NEW CURRENT
  3023    3151              DCA NXLOC               /NXLOC=DOEST'N FOR NEXT TIME
  3024    6224              RIF
  3025    7041              CIA
  3026    1016              TAD INSFLD
  3027    7650              SNA CLA           /IS NEXT SAME AS CURRENT
  3030    5266              JMP SUB1                /YES, TRY NEXT LOWER FIELD


                    /
                    /CHECK FOR ERROR IN NEW FIELD
                    /
  3031    2022      CKERR,  ISZ ERWRD               /POINTER+1
  3032    1201              TAD CMOVE+1
  3033    7041              CIA
  3034    1422              TAD I ERWRD
  3035    7650              SNA CLA           /NO ERRORS RECORDED IF 0
  3036    5310              JMP STMV                /INITIALIZE MOVE

  3037    1422      CNXT,   TAD I ERWRD
  3040    7041              CIA
  3041    1016              TAD INSFLD
  3042    7650              SNA CLA           /ERROR IN NEW FIELD IF 0
  3043    5253              JMP EQUAL
  3044    1022              TAD ERWRD
  3045    7041              CIA
  3046    1021              TAD ENTBL               /ENTBL=ERWRD+10
  3047    7650              SNA CLA           /TABLE DONE IF 0
  3050    5310              JMP STMV                /INITIALIZE MOVE
  3051    2022              ISZ ERWRD               /POINTER+1
  3052    5237              JMP CNXT
```

```
                        /
3053  1422    EQUAL,    TAD I ERWRD     /GET ERROR FIELD
3054  7041              CIA
3055  1123              TAD FIRST1
3056  7650              SNA CLA         /DON'T MOVE IF = TO FIRST
3057  5530              JMP I XRTN      /START OVER
3060  1422              TAD I ERWRD
3061  7650              SNA CLA         /IS IT FIELD 0?
3062  5266              JMP SUB1                /YES
3063  1016              TAD INSFLD              /CURRENT NEXT
3064  1141              TAD M10         /SUBTRACT 1 FROM DF
3065  3151              DCA NXLOC
                        /
3066  1020    SUB1,     TAD ERTBL
3067  3022              DCA ERWRD               /RESTORE TABLE POINTER
3070  1151              TAD NXLOC
3071  7041              CIA
3072  1016              TAD INSFLD
3073  7650              SNA CLA         /NEXT = CURRENT NEXT IF 0
3074  5253              JMP EQUAL
3075  1151              TAD NXLOC
3076  3016              DCA INSFLD      /NEW CURRENT FIELD
3077  1016              TAD INSFLD
3100  7041              CIA
3101  1123              TAD FIRST1      /IS IT = LOWEST FIELD
3102  7650              SNA CLA
3103  5231              JMP CKERR               /YES
3104  1016              TAD INSFLD      /CURRENT NEW FIELD
3105  1141              TAD M10         /SUBTRACT 1 FROM DF
3106  3151              DCA NXLOC               /NEXT FIELD LOWER
3107  5231              JMP CKERR


                        /
3110  7200    STMV,     CLA
3111  1020              TAD ERTBL
3112  3022              DCA ERWRD               /RESTORE TABLE POINTER
3113  6224              RIF
3114  3723              DCA I XSRCE
3115  1723              TAD I XSRCE
3116  7041              CIA
3117  1016              TAD INSFLD
3120  7650              SNA CLA         /DON'T MOVE IF EQUAL
3121  5530              JMP I XRTN      /START OVER
3122  5724              JMP I XMVE      /GO MOVE
                        /
3123  3323    XSRCE,    SOURCE
3124  3307    XMVE,     MOVE
3125  3200    XTMV,     NXTMV
                        /


      3200    *3200
                        /
3200  7600    NXTMV,    7600
```

```
3201  6224            RIF
3202  3323            DCA SOURCE      /CURRENT FIELD
3203  2022  CHNXT,    ISZ ERWRD               /POINTER +1
3204  1200            TAD NXTMV
3205  7041            CIA
3206  1422            TAD I ERWRD
3207  7650            SNA CLA         /NO ERRORS RECORDED IF 0
3210  5225            JMP STNXT               /INITIALIZE MOVE
3211  1422  CKNXT,    TAD I ERWRD
3212  7041            CIA
3213  1151            TAD NXLOC
3214  7650            SNA CLA         /ERROR IN NEW FIELD IF 0
3215  5255            JMP SUB2                /TRY NEXT LOWER FIELD
3216  1022            TAD ERWRD
3217  7041            CIA
3220  1021            TAD ENTBL
3221  7650            SNA CLA         /DONE WITH TABLE IF 0
3222  5225            JMP STNXT               /INITIALIZE MOVE
3223  2022            ISZ ERWRD               /POINTER +1
3224  5211            JMP CKNXT
                /
3225  1020  STNXT,    TAD ERTBL
3226  3022            DCA ERWRD               /RESTORE TABLE POINTER
3227  1151            TAD NXLOC               /NEXT LOWER FIELD
3230  7041            CIA
3231  1016            TAD INSFLD
3232  7650            SNA CLA         /NEXT=CURRENT IF 0
3233  5242            JMP CKNT
3234  1151            TAD NXLOC
3235  7041            CIA
3236  1123            TAD FIRST1
3237  7640            SZA CLA         /NEXT = LOWEST IF 0
3240  5247            JMP STNX
3241  5302            JMP MVBK                /MOVE TO LOWEST TEST FIELD
3242  1151  CKNT,     TAD NXLOC
3243  7041            CIA
3244  1123            TAD FIRST1
3245  7650            SNA CLA         /NEXT = LOWEST IF 0
3246  5275            JMP NXTHI               /SETUP TO MOVE TO HIGHEST
3247  1151  STNX,     TAD NXLOC               /NEXT LOWER FIELD
3250  3016            DCA INSFLD      /IS NOW CURRENT FIELD
3251  1016            TAD INSFLD
3252  1141            TAD M10         /SUBTRACT 1 FROM NEW
3253  3151            DCA NXLOC               /NEW NEXT LOWER FIELD
3254  5307            JMP MOVE                /GO MOVE
                /


                /
3255  1020  SUB2,     TAD ERTBL
3256  3022            DCA ERWRD               /RESTORE TABLE POINTER
3257  1151            TAD NXLOC               /NEXT LOWER FIELD
3260  7450            SNA             /=FIELD 0 IF 0
3261  5530            JMP I XRTN      /START OVER CAN'T MOVE
3262  1141            TAD M10         /SUBTRACT 1
```

```
3263  3151          DCA NXLOC           /NOW = 2 FIELDS LOWER
3264  1151          TAD NXLOC
3265  7041          CIA
3266  1016          TAD INSFLD          /CURRENT FIELD
3267  7640          SZA CLA             /ARE THEY EQUAL
3270  5203          JMP CHNXT           /NO
3271  1151          TAD NXLOC           /YES
3272  7450          SNA                 /DOES IT = FIELD 0
3273  5203          JMP CHNXT           /YES
3274  5262          JMP SUB2+5          /NO
                    /
3275  1124    NXTHI,TAD LAST1           /VERY LAST TO TEST
3276  3151          DCA NXLOC           /MAKE IT NEXT FIELD
3277  1124          TAD LAST1
3300  3016          DCA INSFLD
3301  5203          JMP CHNXT
                    /
3302  1151    MVBK, TAD NXLOC
3303  3016          DCA INSFLD
3304  6224          RIF
3305  3323          DCA SOURCE
3306  3015          DCA FLAGS           /CLEAR BIT 11


                    /
                    /ROUTINE TO RELOCATE 4K FIELDS
                    /
3307  1125    MOVE, TAD KCDF            /6201
3310  1323          TAD SOURCE          /CURRENT FIELD
3311  3323          DCA SOURCE          /SOURCE NOW = CDF N
3312  1125          TAD KCDF            /6201
3313  1016          TAD INSFLD          /NEW FIELD
3314  3327          DCA DESTN           /DESTN NOW = CDF N
3315  1323          TAD SOURCE
3316  7041          CIA
3317  1327          TAD DESTN
3320  7650          SNA CLA
3321  5530          JMP I XRTN
3322  3074          DCA LOOP            /4K COUNTER
3323  0000    SOURCE,0                  /WILL = CDF N
3324  1474          TAD I LOOP          /TAKE FROM HERE
3325  3347          DCA SAVGD           /SAVE INSTRUCTION
3326  1347          TAD SAVGD           /GET IT BACK
3327  0000    DESTN,0
3330  3474          DCA I LOOP          /PUT IN HERE
3331  1474          TAD I LOOP          /GET INFORMATION STORED
3332  7041          CIA
3333  1347          TAD SAVGD           /COMPARE TO THIS VALUE
3334  7650          SNA CLA             /WERE THEY THE SAME
3335  5340          JMP .+3             /YES CONTINUE
3336  7402          HLT                 /NO, RELOCATION ERROR
3337  5323          JMP SOURCE          /TRY SAME AGAIN
3340  2074          ISZ LOOP            /DONE 4K WHEN SKIP
3341  5323          JMP SOURCE          /KEEP MOVING
3342  1126          TAD KCIF            /6202
```

```
3343  1016          TAD INSFLD      /NEW FIELD
3344  3345          DCA .+1
3345  6202          CIF 00          /CHANGE TO NEW FIELD
3346  5530          JMP I XRTN      /EXIT TO RTN1 IN
                                    /NEW FIELD
3347  0000  SAVGD, 0

              $
```

```
0000  01111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
0100  11111111  11111111  11111111  11111111  11111111  11111000  00000000  00000000

0200  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
0300  11111111  11111111  11111111  11111111  11111110  00000000  00000000  00000000

0400  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
0500  11111111  11111111  11110000  00000000  00000000  00000000  00000000  00000000

0600  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
0700  11111111  11111111  11111111  11111111  11111111  11111111  11100000  00000000


1000  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
1100  11111111  11111111  11111111  11111111  11111111  11111000  00000000  00000000

1200  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
1300  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111110

1400  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
1500  11111111  11111111  11111111  11000000  00000000  00000000  00000000  00000000

1600  11111111  11111111  11111111  11111111  11111111  11111111  00000000  00000000
1700  11111111  11111111  11111111  11000000  00000000  00000000  00000000  00000000


2000  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
2100  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11000000

2200  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
2300  11111111  11111111  11111111  11111111  11111111  11110000  00000000  00000000

2400  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
2500  11111111  11111111  11111111  11111111  11111111  11111111  11110000  00000000

2600  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
2700  11111111  11111111  11111111  11111111  11111111  11111110  00000000  00000000


3000  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
3100  11111111  11111111  11111100  00000000  00000000  00000000  00000000  00000000

3200  11111111  11111111  11111111  11111111  11111111  11111111  11111111  11111111
3300  11111111  11111111  11111111  11111111  11111111  00000000  00000000  00000000

3400
3500

3600
3700
```

```
4000
4100

4200
4300

4400
4500

4600
4700


5000
5100

5200
5300

5400
5500

5600
5700


6000
6100

6200
6300

6400
6500

6600
6700


7000
7100

7200
7300

7400
7500

7600
7700
```

ALAW	2220	FLCNT	0073	LEGAL	2314	RD4	1455
ALOK	2273	FLDN	2444	LOCAT	2400	RDALL	0736
BAD	2163	FLOAD	0017	LOOP	0074	RDF	6214
BADD	2634	GENRAN	2716	M10	0141	RDLOP	0753
BEGIN	3200	GODD	2555	M14	0140	RIF	6224
BSPCE	2600	GOOD	2164	M20	0056	RLOPA	1011
CBANK	1607	HALT	2641	M4	0060	RSTRT1	0211
CDF	6201	INSFLD	0016	M40	0057	RTN1	0213
CDON1	2760	JMS1	0110	M5	0061	RXIT	1020
CFLD	1722	JMS2	0111	MCWA	0033	SAVGD	3347
CHAR	0134	JMS3	0112	MEMADR	0122	SAVIF	1623
CHNXT	3203	JMS4	0113	MOVE	3307	SETSR	2645
CIF	6202	JMS5	0114	MVBK	3302	SETU1	2200
CKBNK	1600	K1	0150	NXLOC	0151	SLMTS	2207
CKERR	3031	K10	0034	NXTBNK	1624	SOURCE	3323
CKNT	3242	K100	0040	NXTHI	3275	SPING	2115
CKNXT	3211	K177	2756	NXTMV	3200	STALL	2702
CLOP1	1252	K1K	1036	OCADR	2165	STMV	3110
CLOP2	1327	K20	0042	OKAS	2256	STNX	3247
CLOP3	1405	K200	0037	OTLDR	2541	STNXT	3225
CLOP4	1462	K207	2167	OVER	1706	STSR	2670
CMOVE	3000	K212	2171	PBAD	2605	SUB1	3066
CNXT	3037	K215	0143	PFILD	2500	SUB2	3255
COUNT	0072	K240	2170	PGOOD	2524	SW0	2111
CRLF	2154	K254	2353	PHDR	2474	SW1	2042
DATFLD	0014	K260	0142	PLIMT	2452	SW2	2033
DESTN	3327	K261	0052	PLOCT	2412	TBLRAN	2755
ENTBL	0021	K262	0053	POCDR	2512	TDM20	0106
EPRNT	2054	K263	0054	POSITN	2131	TDM40	0107
EQUAL	3053	K264	0055	PRERR	2146	TLIMT	2446
EREXT	2046	K277	0146	PRGAM	2417	TNUM	0062
ERROR	2000	K370	0145	PROCTL	2126	TST1	0300
ERTBL	0020	K377	0144	PSTSR	2652	TST1C	0324
ERWRD	0022	K40	0041	PTSTN	2617	TST2	0400
EXAM1	0241	K400	0036	QUERY	2344	TST2C	0425
EXAM2	0246	K7	0133	RANCON	2741	TST3	0452
EXAM3	0253	K740	0035	RANDEX	2742	TST3C	0477
EXAM4	0260	KCDF	0125	RANTAB	2754	TST4	0600
EXHDR	2563	KCIF	0126	RANTAD	2734	TST4C	0625
EXIT	0153	KEYIN	2303	RANTBL	2743	TSTL	2457
EXT1	0322	KRXT	1035	RCHK1	1037	TSTN	2626
EXT1C	0343	KXT1	0345	RCHK1C	1054	W0011	0652
EXT2	0423	KXT1C	0346	RCHK2	1071	W0110	0704
EXT2C	0445	KXT2	0450	RCHK2C	1106	W1001	0721
EXT3	0475	KXT2C	0451	RCHK3	1123	W1100	0667
EXT3C	0517	KXT3	0522	RCHK3C	1140	WRLOP	1710
EXT4	0623	KXT3C	0523	RCHK4	1200	WTCR	2660
EXT4C	0647	KXT4	0645	RCHK4C	1217	X0011	0064
FEILD	1646	KXT4C	0646	RCHKA	1000	X0110	0066
FILD	2532	LAST	2166	RD1	1245	X1001	0067
FIRST1	0123	LAST1	0124	RD2	1322	X1100	0065
FLAGS	0015	LASTX	0154	RD3	1400	XBANK	0063

XCFL	1244	XTST3	0046
XCHK1	0076	XTST4	0047
XCHK1C	0077		
XCHK2	0100		
XCHK2C	0101		
XCHK3	0102		
XCHK3C	0103		
XCHK4	0104		
XCHK4C	0105		
XCHKA	1236		
XCRLF	0147		
XFILD	0131		
XHDR	0135		
XHLT	0127		
XIT1	0320		
XIT1C	0341		
XIT2	0421		
XIT2C	0443		
XIT3	0473		
XIT3C	0515		
XIT4	0621		
XIT4C	0643		
XKBNK	0070		
XLCAT	2352		
XLMTS	0043		
XLOPA	1237		
XLOPB	1240		
XLOPC	1241		
XLOPD	1242		
XLOPE	1243		
XMOVE	0050		
XMVE	3124		
XPERR	0136		
XPHDR	2644		
XPING	0137		
XPRER	0132		
XRALL	0075		
XRD1	0115		
XRD2	0116		
XRD3	0117		
XRD4	0120		
XRROR	0121		
XRTN	0130		
XSALL	0152		
XSETU	0051		
XSRCE	3123		
XSTSR	0277		
XTBNK	0071		
XTLIM	2351		
XTMV	3125		
XTST1	0044		
XTST2	0045		

ERRORS DETECTED: 0

LINKS GENERATED: 0

RUN-TIME: 13 SECONDS

2K CORE USED